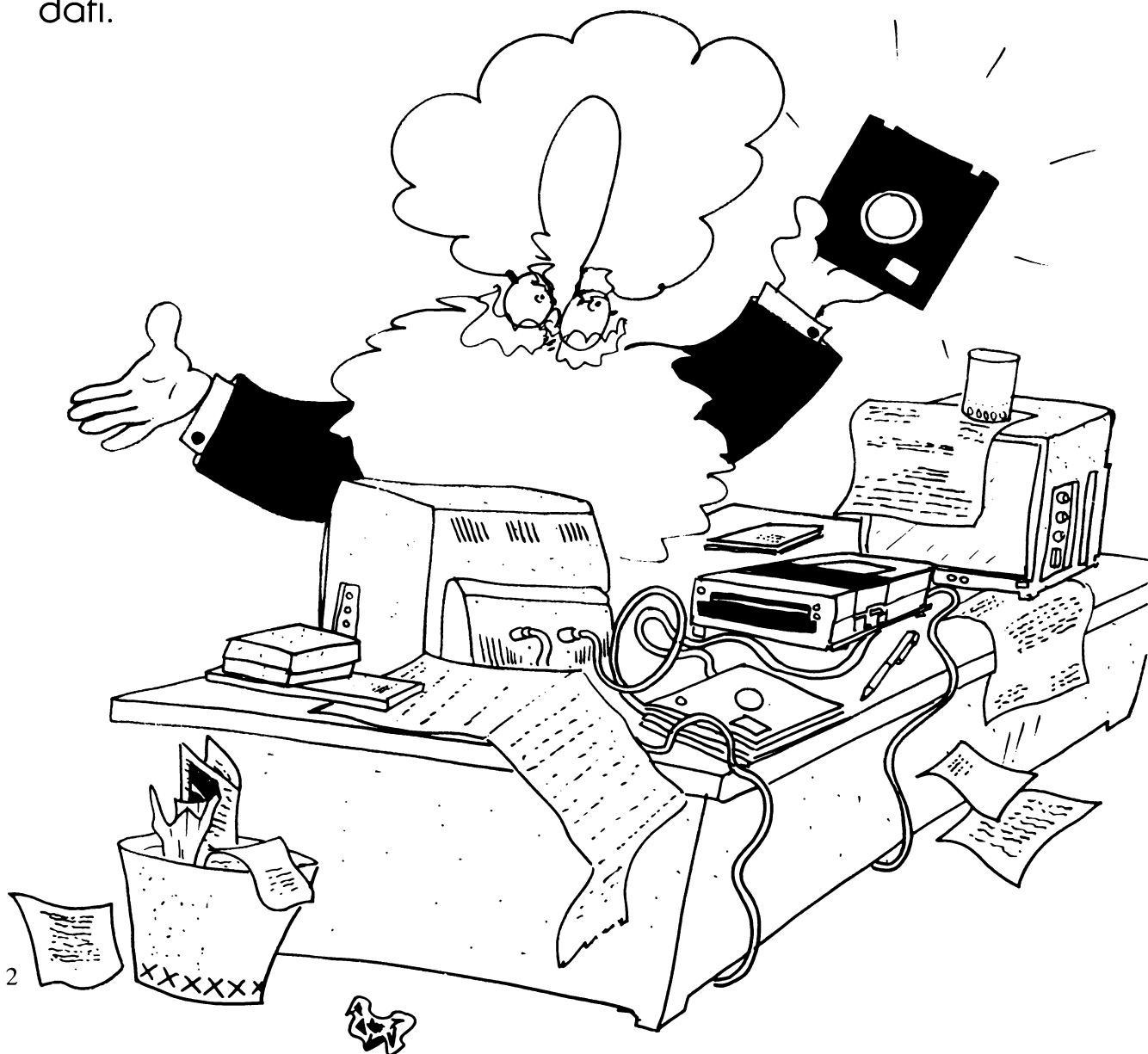
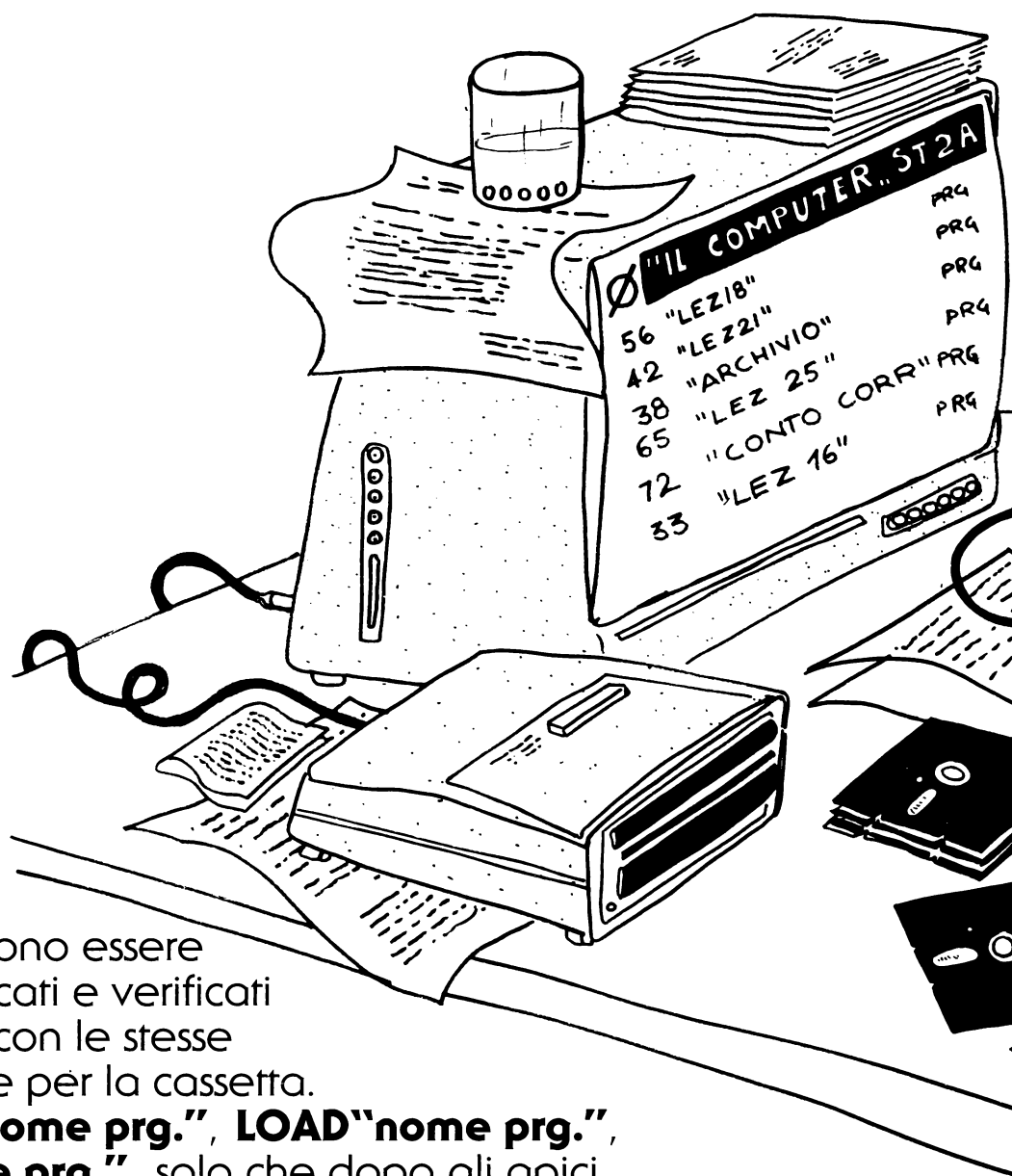
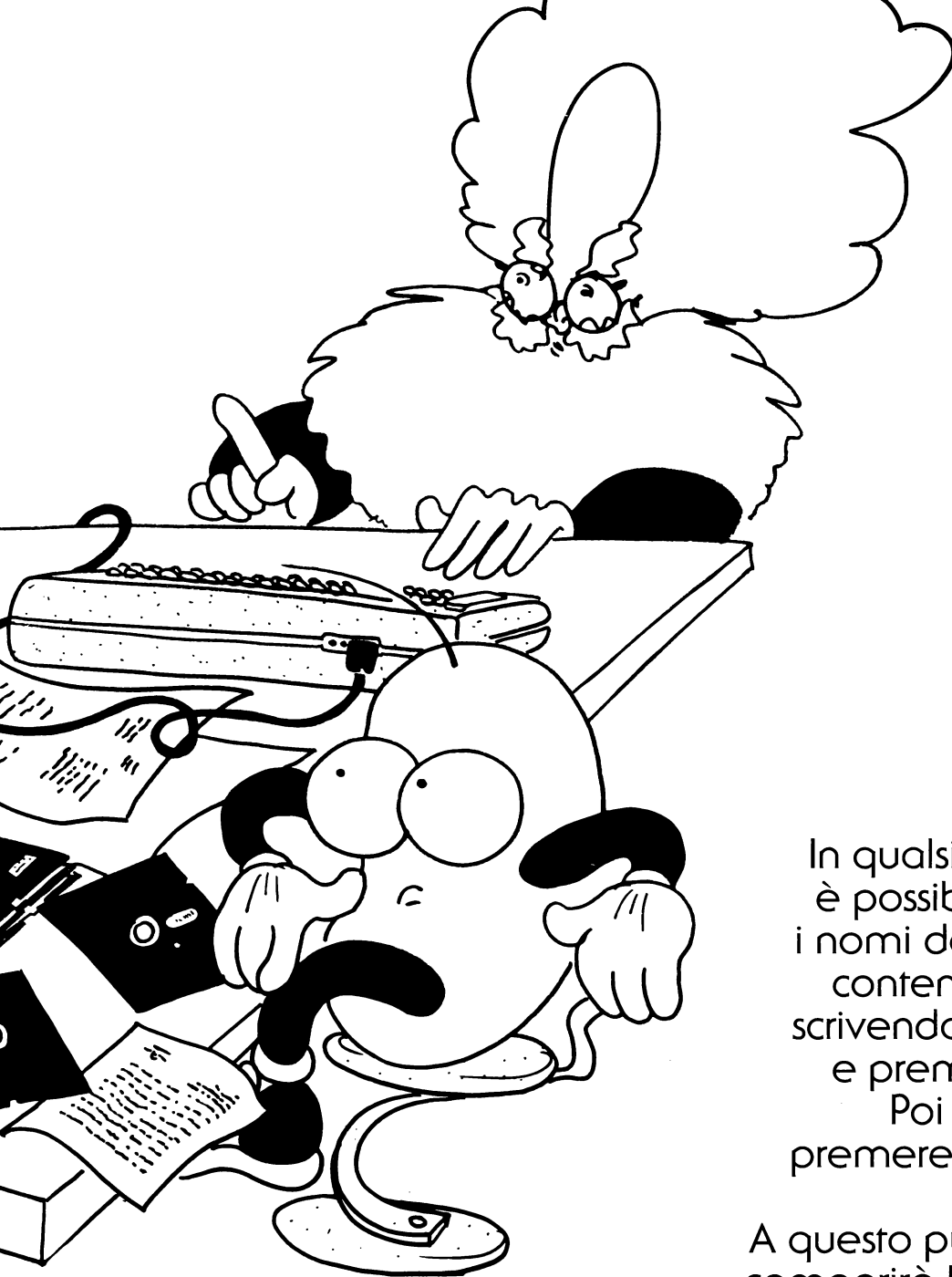
CONOSCERE IL COMPUTER DIRETTAMENTE DAL COMPUTER

per Commodore Vic20 e 64

Beatrice d'Este

Il computer, oltre che al registratore, può anche essere collegato all'unità a disco; in questa lezione ne vedremo l'utilizzo.
Il disco svolge la stessa funzione del registratore; cioè conserva programmi e dati anche quando il computer viene spento.
I vantaggi del disco in rapporto al registratore sono:
1) Una velocità di trasferimento dati 10 volte maggiore
2) Il nastro per essere letto o registrato deve essere posizionato manualmente, mentre l'unità a disco ricerca automaticamente i programmi o gli spazi ancora liberi di memoria.
3) Una maggiore affidabilità di lettura e di registrazione dei dati.

La prima operazione da compiere per l'utilizzo di un disco è quella di formattarlo, visto che all'inizio il disco è vergine. La **FORMATTAZIONE** serve per suddividere il disco in tracce e settori che serviranno al computer per potersi orientare a rintracciare i dati. Ricorda di non formattare mai un disco già inciso, perchè in questo modo verrebbero cancellati tutti i dati già memorizzati.
Per formattarlo, quindi, basta inserirlo nel drive e scrivere il seguente comando: **OPEN1,8,15, "N:nome disco,id"** poi battere il tasto Return e infine, quando il led rosso del drive si è spento, scrivere: **CLOSE1** e premere il Return.
Compiuta la formattazione il disco è pronto per l'uso.

Sul disco possono essere registrati, caricati e verificati i programmi con le stesse istruzioni usate per la cassetta.
Cioè: **SAVE"nome prg."**, **LOAD"nome prg."**, **VERIFY"nome prg."**, solo che dopo gli apici deve essere aggiunto **, 8** per indicare al computer di compiere l'operazione sul disco e non sulla cassetta.
Così, ad esempio, per registrare su disco un programma basterà scrivere: **SAVE"nome prg.",8**
Ricorda che nell'uso del disco è obbligatorio mettere il nome del programma, questo servirà al computer per poterlo ricercare.
Naturalmente sullo stesso disco non potranno esistere due programmi con lo stesso nome, altrimenti il computer non saprebbe a quale riferirsi.

In qualsiasi momento
è possibile conoscere
i nomi dei programmi
contenuti in un disco
scrivendo: **LOAD"$",8**
e premere il Return.
Poi scrivere **LIST** e
premere nuovamente
il Return.
A questo punto sul video
comparirà la **DIRECTORY**
(o catalogo) del dischetto,
cioè l'elenco di tutti i
programmi memorizzati sul disco.
Il numero a sinistra di ogni nome indicherà la lunghezza, in
blocchi, del programma corrispondente. Nell'ultima riga
invece verrà mostrato il numero dei blocchi ancora liberi.
Tieni presente che ogni blocco corrisponde a 256 bytes (1/4
di K) e che un **floppy** contiene 664 blocchi, cioè circa 170 K
di memoria.

Listato dell'esercizio: ESEMPIO PRATICO (CBM 64)

```
10 poke53280,6:poke53281,6
20 v=1024:c=55296:t=3:q=40:l=24
30 print"♥"
40 forx=0tol
50 ifx/t<>int(x/t)then100
60 fory=0toq-1
70 p=x*q+y
80 pokev+p,160:pokec+p,7
90 nexty
100 fory=0toq-1stept
110 p=x*q+y
120 pokev+p,160:pokec+p,7
130 nexty
140 nextx
150 foru=1to3000:nextu
160 print"♥"
```

Listato dell'esercizio: ESEMPIO PRATICO (VIC 20)

```
10 poke36879,110
20 v=7680:c=38400:t=3:q=22:l=21
30 print"♥"
40 forx=0tol
50 ifx/t<>int(x/t)then100
60 fory=0toq-1
70 p=x*q+y
80 pokev+p,160:pokec+p,7
90 nexty
100 fory=0toq-1stept
110 p=x*q+y
120 pokev+p,160:pokec+p,7
130 nexty
140 nextx
150 foru=1to3000:nextu
160 print"♥"
```

Listato dell'esercizio: COMPLETA IL LISTATO (CBM 64)

```
10 poke53280,2:poke53281,2
20 printchr$(147):[*]=1024:c=55296
30 forx=[*]to19
40 [*]=x*40+19-x
50 pokev+p,223:pokec+p,7
60 if[*]=0theny=20:goto110
70 [*]y=20-xto19+x
80 p=x*40+[*]
90 pokev+p,[*]:pokec+p,7
100 [*]y
110 p=x*40+y:pokev+p,223:pokec+p,[*]
120 nextx
130 forv=1to5:for[*]to6
140 [*]53280,s:poke53281,s
150 for[*]to200:nextk
160 nexts,v:end
```

Listato dell'esercizio: COMPLETA IL LISTATO (VIC 20)

```
10 poke36879,42
20 printchr$(147):[*]=7680:c=38400
30 forx=[*]to10
40 [*]=x*22+10-x
50 pokev+p,223:pokec+p,7
60 if[*]=0theny=11:goto110
70 [*]y=11-xto10+x
80 p=x*22+[*]
90 pokev+p,[*]:pokec+p,7
100 [*]y
110 p=x*22+y:pokev+p,223:pokec+p,[*]
120 nextx
130 forv=1to5:for[*]to6
140 [*]36879,s*17+8
150 for[*]to200:nextk
160 nexts,v:end
```

PROGRAMMIAMO INSIEME (CBM 64)

```
10 poke53280,7:poke53281,7
20 print"■■ARCHIVIO FILM"
100 print"■■TITOLO (max 20 car.):■■"
110 l=20:gosub3000:n$=a$:print
200 print"■■REGISTA(max 18 car.):■■"
210 l=18:gosub3000:v$=a$:print
300 print"■■DISTR. (max 14 car.):■■"
310 l=14:gosub3000:c$=a$:print
400 print"■■DATI INSERITI:■■"
420 printn$:printv$:printc$
500 end
3000 l=l+1:s$="":a$="":forf=1tol:s$=s$+"█":a$=a$+" ":nextf
3010 q=1:print"■"left$(a$,1)"■█";
3020 getz$:ifz$=chr$(13)thenreturn
3040 if(z$="█")*(q<l)then4000
3050 if(z$="█")*(q>1)then4200
3060 if(z$=chr$(20))*(q>1)then4400
3070 if(z$=chr$(148))*(right$(a$,2)="  ")then4600
3080 ifz$=chr$(34)thenz$="'"
3100 if((z$>=" ")*(z$<="←"))+((z$>="A")*(z$<="Z"))then3200
3120 goto3020
3200 ifq>l-1then3020
3400 a$=left$(a$,q-1)+z$+right$(a$,l-q)
:q=q+1:printz$"■"mid$(a$,q,1)"■█";
3500 goto3020
4000 q=q+1:printmid$(a$,q-1,1)"■"mid$(a$,q,1)"■█";:goto3020
4200 q=q-1:print"█■"mid$(a$,q,1)"■"mid$(a$,q+1,1)"███";:goto3020
4400 printz$"■"mid$(a$,q,1)"■█";:q=q-1:a$=left$(a$,q-1)
+right$(a$,l-q)+" "
4420 goto3020
4600 printmid$(a$,q,1)"█■ ■"z$mid$(a$,q,1)"███";
4620 a$=left$(a$,q-1)+" "+mid$(a$,q,l-q)
4640 goto3020
```

PROGRAMMIAMO INSIEME (VIC 20)

Il listato è identico a quello per il CBM 64, tranne nella linea:

```
10 poke36879,127
```

Soluzione dell'esercizio: COMPLETA IL LISTATO CBM 64

(lez. 27)

```
10 poke53280,6:poke53281,3
20 n=0:printchr$(147)
30 l1=24:l2=38:x=0:y=1
40 forg=1to6
50 l1=l1-2:forx=xtox+l1
60 gosub150:nextx
70 l2=l2-2:fory=ytoy+l2
80 gosub150:nexty
90 ifg=1thenl1=l1+1
100 l1=l1-2:forx=xtox-l1step-1
110 gosub150:nextx
120 l2=l2-2:fory=ytoy-l2step-1
130 gosub150:nexty
140 nextg:end
150 p=x*40+y:poke55296+p,2
160 poke1024+p,160:return
```

Soluzione dell'esercizio: COMPLETA IL LISTATO VIC 20

(lez. 27)

```
10 poke36879,62
20 n=0:printchr$(147)
30 l1=22:l2=20:x=0:y=1
40 forg=1to5
50 l1=l1-2:forx=xtox+l1
60 gosub150:nextx
70 l2=l2-2:fory=ytoy+l2
80 gosub150:nexty
90 ifg=1thenl1=l1+1
100 l1=l1-2:forx=xtox-l1step-1
110 gosub150:nextx
120 l2=l2-2:fory=ytoy-l2step-1
130 gosub150:nexty
140 nextg:end
150 p=x*22+y:poke38400+p,2
160 poke7680+p,160:return
```